Computers and Mathematics with Applications 61 (2011) 3506–3518

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

An immersed boundary method based on the lattice Boltzmann approach in three dimensions, with application

Luoding Zhu [a,*], Guowei He [b], Shizhao Wang [b], Laura Miller [c], Xing Zhang [b], Qian You [d], Shiaofen Fang [d]

[a] Department of Mathematical Sciences, Indiana University–Purdue University Indianapolis, Indianapolis, IN 46202, United States
[b] State Key Laboratory of Nonlinear Mechanics, Institute of Mechanics, Chinese Academy of Sciences, Beijing, 10088, PR China
[c] Department of Mathematics, University of North Carolina Chapel Hill, Chapel Hill, NC 27599, United States
[d] Department of Computer Science, Indiana University–Purdue University Indianapolis, Indianapolis, IN 46202, United States

ARTICLE INFO

Keywords:
Immersed boundary method
Lattice Boltzmann method
Fluid–structure interaction
Deformable body
Drag scaling
Computational fluid dynamics
Incompressible viscous flow

ABSTRACT

The immersed boundary (IB) method originated by Peskin has been popular in modeling and simulating problems involving the interaction of a flexible structure and a viscous incompressible fluid. The Navier–Stokes (N–S) equations in the IB method are usually solved using numerical methods such as FFT and projection methods. Here in our work, the N–S equations are solved by an alternative approach, the lattice Boltzmann method (LBM). Compared to many conventional N–S solvers, the LBM can be easier to implement and more convenient for modeling additional physics in a problem. This alternative approach adds extra versatility to the immersed boundary method. In this paper we discuss the use of a 3D lattice Boltzmann model (D3Q19) within the IB method. We use this hybrid approach to simulate a viscous flow past a flexible sheet tethered at its middle line in a 3D channel and determine a drag scaling law for the sheet. Our main conclusions are: (1) the hybrid method is convergent with first-order accuracy which is consistent with the immersed boundary method in general; (2) the drag of the flexible sheet appears to scale with the inflow speed which is in sharp contrast with the square law for a rigid body in a viscous flow.

1. Introduction

Problems involving fluid–flexible-structure interactions are ubiquitous in Nature: fish swimming in the ocean [1], sessile marine organisms waving in the water [2], and red blood cells moving in the flowing blood in human vessels [3], to name but a few. These problems involve the complicated interplay between a viscous fluid, deformable body, and free moving boundary, making them difficult to discern. Consequently, analytic solutions are rare and almost nonexistent, but a computational approach is a viable possibility.

Currently there exist a variety of methods developed for handling this type of problem. Traditionally solving a fluid–structure-interaction (FSI) problem with complex geometry relies on the body-fitted unstructured grid method [4] or the overset grid method [5]. A somewhat non-conventional approach, the immersed boundary (IB) method originated by Charles Peskin [6,7] in 1972, was probably the first method developed for addressing the fully fluid–flexible-structure-interaction problem. Since the birth of the IB method, many other methods for fluid–structure-interaction problems have emerged. These include the arbitrary Lagrangian Eulerian method [8,9], the material point method [10,11], the fictitious

* Corresponding author. Tel.: +1 317 278 9225; fax: +1 317 274 3460.
E-mail address: lzhu@math.iupui.edu (L. Zhu).

0898-1221/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2010.03.022

domain method [12–14], the immersed interface method [15–18], the ghost fluid method [19–21], the immersed finite element method [22,23], the extended immersed boundary method [24], the immersed continuum method [25,26], the level set method [27–30], and other IB methods designed specifically for fluid–*rigid*-body interactions [31–34].

Various versions of the IB method have been developed in response to the needs of their application: in addition to the original version [35,6,36–38], there exist the vortex-method version [39], the volume-conserved version [40,41], the adaptive mesh version [42], the (formally) second-order versions [43–45], the multigrid version [46], the penalty version [47], the implicit versions [48–50,16,51,52], and the stochastic version [53,54].

A basic element of the IB method is how to solve the viscous incompressible flow problem. Usually, the Navier–Stokes equations are solved using a variety of numerical techniques, such as the fast Fourier transform, the projection method, and the particle-in-cell method. Another novel and popular approach is to replace the Navier–Stokes equations solver by the lattice Boltzmann method (LBM).

The LBM is an alternative to traditional numerical methods for obtaining the flow solution. Rather than solving for the macroscopic variables such as velocity and pressure directly, the LBM employs a mesoscopic description and deals with a single-particle velocity distribution function $g(\mathbf{x}, \boldsymbol{\xi}, t)$ ($\mathbf{x}$ represents the spatial coordinate, $\boldsymbol{\xi}$ is the particle velocity, and t is time) that obeys an approximate Boltzmann equation. It has been well established in the past few decades that the LBM is a powerful tool of computational fluid dynamics (CFD) [55–64] and is applicable to almost all branches of CFD [65]. Compared to conventional approaches for solving the flow problem, the LBM is simpler to use, easier to parallelize, and more convenient to use to incorporate additional physics into a model to simulate new phenomena, particularly in three dimensions.

In this paper we discuss a three-dimensional IB method using the lattice Boltzmann (LB) approach (the D3Q19 model) for obtaining the viscous incompressible flow in simulating the interaction in a 3D viscous flow past a flexible sheet tethered at the middle line. The aim of this study is to focus on drag scaling laws for flexible sheets.

Very recently a combination of the IB method and the LBM (the IB–LBM) was introduced to simulate the interaction of *rigid* particles with viscous incompressible flows in the area of particulate flows by Feng and Michaelides [66,67]. An improvement of the IB–LBM in 2D was proposed by Niu et al. [68] where a modified momentum exchange method [69] was used for computing the immersed boundary force at the known rigid boundary position. (The methods used in [66,67] were the penalty method and the direct forcing scheme.) Another improvement of the IB–LBM [66,67] in 2D was proposed by Sui et al. [70] where the authors used the multi-block version of the LBM [71,72] for the coupling with the IB method. This hybrid IB and multi-block LB method is highly suitable for treating high *Re* flows. Peng and Luo [73] did a comparative study on the immersed boundary method and the lattice Boltzmann method in handling fluid–structure interaction in two dimensions. Other hybrid methods involving the immersed boundary and the lattice Boltzmann method include those of Le and Zhang [74], Sui, Chew, et al. [75], Feng and Michaelides [76], and Cheng and Zhang [77]. Note that the works [67,75,76] are three dimensional while the remainder are two dimensional.

In this paper, we discuss a three-dimensional IB method using the D3Q19 lattice Boltzmann model to provide the flow solution and adopting the approach of Guo et al. [78] to treat the external forces (including the immersed boundary force) within the LBM. We note that this treatment does not alter the way to compute the immersed boundary force.

Our approach is similar to the work by Sui et al. [75]. The major differences between our method and some other existing hybrid IB/LBM methods are as follows:

(1) The IB/LBM of Feng and Michaelides [66,67] is formulated for rigid particles; our work is formulated for flexible structures. The rigid particles' motions are governed by Newton's second law: $\mathbf{F} = m\mathbf{a}$. In contrast, the motion of a flexible structure in our study is described by $\frac{\partial \mathbf{X}}{\partial t} = \mathbf{U}$, and Newton's second law is used to compute the force that the immersed structure exerts on the ambient fluid.

(2) The external forcing term in the LBM is treated differently. The approach to include the immersed boundary force in the IB–LBM results in a scheme which does not converge to the incompressible viscous Navier–Stokes equations in general. It does so only under some assumptions which cannot hold true in the IB context [78]. We use another way to introduce external forces into the LBM which has been studied and shown by Guo et al. [78] to recover the viscous incompressible Navier–Stokes equations unconditionally.

We apply our newly developed 3D IB/LBM method to study via simulation viscous flows in a 3D channel past a flexible sheet fixed at its midline. This problem generalizes an important 2D problem of viscous flow past an elastic fiber tethered at its center point treated by Steinberg [79], by Alben, Shelley and Zhang [80,81], and by Zhu and Peskin [82–84]. The 3D extension is significant. Classic theory [85] predicts that the drag of a rigid body in a rapidly flowing medium is proportional to the square of the oncoming medium speed. This law may not hold for a flexible body because it is compliant and accommodates the flow by bending and streamlining. Very recently Alben, Shelley and Zhang have found that the drag of a flexible fiber in a 2D flowing soap film scales with a factor of $\frac{4}{3}$ of the incoming flow speed at very high Reynolds numbers [80,81]. Zhu [84] has found that the exponent of the drag scaling depends on the Reynolds number: the exponent decreases monotonically from approximately 2 towards 4/3 as *Re* increases from 10 to 800. What can go on in three dimensions? There are two important dimensionless parameters in fluid–flexible-structure-interaction problems: the Reynolds number and the dimensionless bending modulus (see Table 1 for definitions). How do these dimensionless parameters influence the drag scaling in three dimensions? We attempt to answer these questions via numerical simulations. Our major result is

that the drag of the flexible sheet appears to scale as the first power of the inflow speed in the ranges of the dimensionless parameters used in our simulations.

The remainder of the paper is as follows. A general 3D IB/LBM formulation is described in Section 2 including governing equations and their discretization. Section 3 describes the modeling of the application problem and the major simulation results including method validation, flow visualization and drag scaling. Section 4 concludes the paper with a summary and discussion.

2. The lattice Boltzmann IB formulation

2.1. Governing equations

Choosing appropriate reference quantities for length (the width of the plate W), velocity (the inflow speed U) and mass density (the constant fluid mass density ρ_0), our lattice Boltzmann IB formulation for fluid–flexible-structure interaction is formulated in dimensionless form as follows:

$$\frac{\partial g(\mathbf{x},\boldsymbol{\xi},t)}{\partial t}+\boldsymbol{\xi}\cdot\frac{\partial g(\mathbf{x},\boldsymbol{\xi},t)}{\partial \mathbf{x}}+\mathbf{f}(\mathbf{x},t)\cdot\frac{\partial g(\mathbf{x},\boldsymbol{\xi},t)}{\partial \boldsymbol{\xi}}=-\frac{1}{\tau}(g(\mathbf{x},\boldsymbol{\xi},t)-g^{(0)}(\mathbf{x},\boldsymbol{\xi},t)), \tag{1}$$

Eq. (1) is the Bhatnagar–Gross–Krook (BGK) equation [86] which is used to describe the motion of both the fluid and the immersed boundary. The function $g(\mathbf{x},\boldsymbol{\xi},t)$ is the single-particle velocity distribution function, where $\mathbf{x}$ is the spatial coordinate, $\boldsymbol{\xi}$ is the particle velocity, and t is time. $f(\mathbf{x},\boldsymbol{\xi},t)\mathrm{d}\mathbf{x}\mathrm{d}\boldsymbol{\xi}$ represents the probability of finding a particle at time t located in $[\mathbf{x},\mathbf{x}+\mathrm{d}\mathbf{x}]$ moving with a velocity between $\boldsymbol{\xi}$ and $\boldsymbol{\xi}+\mathrm{d}\boldsymbol{\xi}$. The term $-\frac{1}{\tau}(g-g^{(0)})$ is the well-known BGK approximation [86] to the complex collision operator in the Boltzmann equation, where the τ is the relaxation time (dimensionless). It is connected to the fluid kinematic viscosity ν (dimensionless) in the LBM. (Different models may have different τ–ν relationships. For the D3Q19 model, $\nu=\frac{2\tau-1}{6}$.) The $g^{(0)}$ is the Maxwellian distribution. The external force term $\mathbf{f}(\mathbf{x},t)=\mathbf{f}_{ib}(\mathbf{x},t)+\mathbf{f}_{ext}(\mathbf{x},t)$. The $\mathbf{f}_{ib}(\mathbf{x},t)$ is the force imparted by the immersed boundary to the fluid. The $\mathbf{f}_{ext}(\mathbf{x},t)$ is other external forces acting on the fluid, e.g. the gravity. The macroscopic variables such as the fluid mass density (ρ) and momentum ($\rho\mathbf{u}$) can be computed from the velocity distribution function g via Eqs. (2) and (3).

$$\rho(\mathbf{x},t)=\int g(\mathbf{x},\boldsymbol{\xi},t)\mathrm{d}\boldsymbol{\xi}, \tag{2}$$

$$(\rho\mathbf{u})(\mathbf{x},t)=\int g(\mathbf{x},\boldsymbol{\xi},t)\boldsymbol{\xi}\mathrm{d}\boldsymbol{\xi}. \tag{3}$$

The Eulerian force density $\mathbf{f}_{ib}(\mathbf{x},t)$ defined on the fixed Eulerian lattice is calculated from the Lagrangian force density $\mathbf{F}(\boldsymbol{\alpha},t)$ defined on the Lagrangian grid by Eq. (4).

$$\mathbf{f}_{ib}(\mathbf{x},t)=\int \mathbf{F}(\boldsymbol{\alpha},t)\delta(\mathbf{x}-\mathbf{X}(\boldsymbol{\alpha},t))\mathrm{d}\boldsymbol{\alpha} \tag{4}$$

where the function $\delta(\mathbf{x})$ is the Dirac δ-function. The Lagrangian force density $\mathbf{F}$ is computed as follows:

$$\mathbf{F}(\boldsymbol{\alpha},t)=-\frac{\partial \mathcal{E}}{\partial \mathbf{X}}=-\frac{\partial(\mathcal{E}_s+\mathcal{E}_b)}{\partial \mathbf{X}}. \tag{5}$$

In Eq. (5) the elastic potential energy density ($\mathcal{E}$) consisted of a stretching/compression part ($\mathcal{E}_s$) and a bending ($\mathcal{E}_b$) part:

$$\mathcal{E}=\mathcal{E}_s+\mathcal{E}_b$$

and these are defined respectively by Eqs. (6) and (7),

$$\mathcal{E}_s=\frac{1}{2}K_s\int\int \mathrm{d}\alpha_2\mathrm{d}\alpha_3\int\left(\left|\frac{\partial \mathbf{X}}{\partial \alpha_1}\right|-1\right)^2\mathrm{d}\alpha_1 \tag{6}$$

$$\mathcal{E}_b=\frac{1}{2}K_b\int\int \mathrm{d}\alpha_2\mathrm{d}\alpha_3\int\left|\frac{\partial^2\mathbf{X}(\boldsymbol{\alpha},t)}{\partial {\alpha_1}^2}\right|^2\mathrm{d}\alpha_1. \tag{7}$$

The variables $\alpha_1,\alpha_2,\alpha_3$ are the three components of the Lagrangian variable $\boldsymbol{\alpha}$. In the case of an immersed surface, such as the plate in our application problem (see Section 3 for details), α_2 may be used to denote a fiber, α_1 to denote the arc-length along the fiber at its initial configuration, and α_3 is not used. Note that in this case we assume that the elastic plate is made of intersecting longitudinal and lateral fibers. K_s is the stretching/compression coefficient and constant K_b is the bending rigidity. Both constants are related to the Young's modulus of the plate.

The motion of the immersed plate is described by a system of first-order ordinary differential equations, Eq. (8):

$$\frac{\partial \mathbf{X}}{\partial t}(\boldsymbol{\alpha},t)=\mathbf{U}(\boldsymbol{\alpha},t). \tag{8}$$

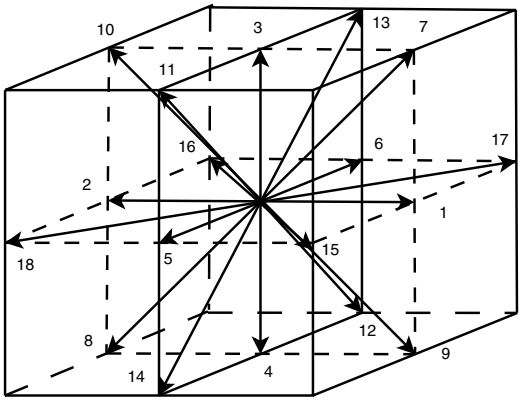

Fig. 1. Lattice Boltzmann D3Q19 model. Particles at a typical node (the center of the cube) can move along 18 different directions, denoted by 1 through 18. Particles are also allowed to stay at the node.

The $\mathbf{X}(\boldsymbol{\alpha}, t)$ is the Eulerian coordinate of the immersed plate at time t whose Lagrangian coordinate is $\boldsymbol{\alpha}$. The immersed boundary velocity $\mathbf{U}(\boldsymbol{\alpha}, t)$ is interpolated from the fluid velocity $\mathbf{u}(\mathbf{x}, t)$ by using the same δ-function to impart the boundary force to the fluid (see Eq. (9)):

$$\mathbf{U}(\boldsymbol{\alpha}, t) = \int \mathbf{u}(\mathbf{x}, t)\delta(\mathbf{x} - \mathbf{X}(\boldsymbol{\alpha}, t))\mathrm{d}\mathbf{x}. \tag{9}$$

2.2. Discretization

The above non-linear system of differential–integral equations (Eqs. (1)–(9)) is discretized on a uniform fixed Eulerian cubic lattice for the fluid with the uniform meshwidth h (the number of grid nodes is N_x, N_y and N_z in the x, y and z directions, respectively), plus a collection of moving Lagrangian discrete points for the immersed boundary with meshwidth $\Delta\alpha_1 = \Delta\alpha_2 = \frac{1}{2}h$. The D3Q19 model [55,56] is used to discretize the BGK equation (Eq. (1)). It has been shown that the D3Q19 model is better than the D3Q15 and D3Q27 models in terms of computational reliability and efficiency [87]. In the D3Q19 model, particles impinging on and exiting at each lattice node can move along eighteen different directions. They are also allowed to stay together with the rest state at the node. Thus, the particle velocity space $\boldsymbol{\xi}$ is discretized by a finite set of 19 velocities. The discrete velocity can be written as (see Fig. 1 for a diagram)

$$\boldsymbol{\xi}_j = \begin{cases} (0, 0, 0), & j = 0 \\ (\pm 1, 0, 0), (0, \pm 1, 0), (0, 0, \pm 1), & j = 1, 2, \ldots, 6 \\ (\pm 1, \pm 1, 0), (\pm 1, 0, \pm 1), (0, \pm 1, \pm 1), & j = 7, 8, \ldots, 18. \end{cases}$$

Let $g_j(\mathbf{x}, t)$ be the distribution function along $\boldsymbol{\xi}_j$. A second-order space and time discretization in a Lagrangian coordinate system is applied to derive the lattice Boltzmann equation (LBE) that advances $g_j(\mathbf{x}, t)$ forward by one step (the dimensionless time step size is set to 1 in the LBM used here):

$$g_j(\mathbf{x} + \boldsymbol{\xi}_j, t + 1) = g_j(\mathbf{x}, t) - \frac{1}{\tau}(g_j(\mathbf{x}, t) - g_j^0(\mathbf{x}, t)) + \left(1 - \frac{1}{2\tau}\right) w_j \left(\frac{\boldsymbol{\xi}_j - \mathbf{u}}{{c_s}^2} + \frac{\boldsymbol{\xi}_j \cdot \mathbf{u}}{{c_s}^4}\boldsymbol{\xi}_j\right) \cdot \mathbf{f} \tag{10}$$

where w_j is the weight, which takes the values

$$w_j = \begin{cases} 1/3, & j = 0 \\ 1/18, & j = 1, 2, \ldots, 6 \\ 1/36, & j = 7, 8, \ldots, 18. \end{cases}$$

Here $c_s = c/\sqrt{3}$ is speed of sound of the model, and c is the lattice speed associated with the D3Q19 model:

$$c = \begin{cases} 0, & j = 0 \\ 1, & j = 1, 2, \ldots, 6 \\ \sqrt{2}, & j = 7, 8, \ldots, 18. \end{cases}$$

In this model, the relaxation time τ is related to the dimensionless fluid viscosity ν by the equation $\nu = \frac{2\tau - 1}{6}$. We follow Guo et al. [78] in treating the external forces of Eq. (10). They proved that their treatment is more accurate than the one used in the LB/LBM [66,67]. Here the fluid velocity $\mathbf{u}$ and forces $\mathbf{f}_{ib}$ and $\mathbf{f}_{ext}$ are evaluated at time t.

The macroscopic variables such as density $\rho(\mathbf{x}, t)$ and momentum $\rho\mathbf{u}(\mathbf{x}, t)$ are related to the $g_j(\mathbf{x}, t)$ at each node by

$$\rho(\mathbf{x}, t) = \sum_j g_j(\mathbf{x}, t), \tag{11}$$

$$(\rho\mathbf{u})(\mathbf{x}, t) = \sum_j \boldsymbol{\xi}_j g_j(\mathbf{x}, t) + \frac{\mathbf{f}(\mathbf{x}, t)}{2}. \tag{12}$$

For an isothermal fluid, the equilibrium distribution function g_j^0 (which is a function of ρ and $\mathbf{u}$) in the D3Q19 model is given by

$$g_j^0(\mathbf{x}, t) = \rho(\mathbf{x}, t) w_j \left(1 + 3\boldsymbol{\xi}_j \cdot \mathbf{u}(\mathbf{x}, t) + \frac{9}{2}(\boldsymbol{\xi}_j \cdot \mathbf{u}(\mathbf{x}, t))^2 - \frac{3}{2}\mathbf{u}(\mathbf{x}, t) \cdot \mathbf{u}(\mathbf{x}, t)\right). \tag{13}$$

The bounce-back scheme [88] is used to model the no-slip boundary condition for a *fixed* rigid wall (four sidewalls in our problem). Notice that no special treatment for the freely moving immersed flexible boundary is needed on the LBM part. It is handled by the IB method through the immersed boundary force. The LBE "feels" the existence of the immersed flexible boundary through the force.

The duration of the time step is set to 1 in the LBM; let n be the time step index: $g^n = g(\mathbf{x}, \boldsymbol{\xi}, n)$, $\mathbf{X}^n(\boldsymbol{\alpha}) = \mathbf{X}(\boldsymbol{\alpha}, n)$, $\mathbf{u}^n = \mathbf{u}(\mathbf{x}, n)$, $p^n = p(\mathbf{x}, n)$, $\rho^n = \rho(\mathbf{x}, n)$. Let the plate be represented by a discrete collection of fibers whose Lagrangian coordinate is α_2. Let $\alpha_2 = q\Delta\alpha_2$, where q is an integer. Let each fiber be represented by a discrete collection of points whose Lagrangian coordinate is α_1. Let $\alpha_1 = m\Delta\alpha_1$, where m is an integer. The "half-integer" points are given by $\alpha_1 = (m + 1/2)\Delta\alpha_1$. For any function $\phi(\alpha)$, let

$$(D_\alpha\phi)(\alpha) = \frac{\phi(\alpha + \frac{\Delta\alpha}{2}) - \phi(\alpha - \frac{\Delta\alpha}{2})}{\Delta\alpha}. \tag{14}$$

Then the stretching energy and the corresponding force are discretized as the following:

$$\mathcal{E}_s = \frac{1}{2}K_s \sum_m (|D_{\alpha_1}\mathbf{X}| - 1)^2 \Delta\alpha_1 = \frac{1}{2}K_s \sum_{m=1}^{n_f - 1} \left(\frac{|\mathbf{X}_{m+1} - \mathbf{X}_m|}{\Delta\alpha_1} - 1\right)^2 \Delta\alpha_1 \tag{15}$$

$$(\mathbf{F}_s)_l = \frac{K_s}{{\Delta\alpha_1}^2} \sum_{m=1}^{n_f - 1} (|\mathbf{X}_{m+1} - \mathbf{X}_m| - \Delta\alpha_1) \frac{\mathbf{X}_{m+1} - \mathbf{X}_m}{|\mathbf{X}_{m+1} - \mathbf{X}_m|} (\delta_{ml} - \delta_{m+1,l}). \tag{16}$$

Here $(\mathbf{F}_s)_l$, $l = 1, 2, \ldots, n_f$, is the Lagrangian force density $\mathbf{F}_s$ associated with the node l. The bending energy and the corresponding force are discretized as follows:

$$\mathcal{E}_b = \frac{1}{2}K_b \sum_m |D_{\alpha_1} D_{\alpha_1} \mathbf{X}|^2 \Delta\alpha_1 = \frac{1}{2}K_b \sum_{m=2}^{n_f - 1} \left[\frac{|\mathbf{X}_{m+1} + \mathbf{X}_{m-1} - 2\mathbf{X}_m|^2}{(\Delta\alpha_1)^4}\right] \Delta\alpha_1 \tag{17}$$

$$(\mathbf{F}_b)_l = \frac{K_b}{(\Delta\alpha_1)^4} \sum_{m=2}^{n_f - 1} (\mathbf{X}_{m+1} + \mathbf{X}_{m-1} - 2\mathbf{X}_m)(2\delta_{ml} - \delta_{m+1,l} - \delta_{m-1,l}). \tag{18}$$

Here $(\mathbf{F}_b)_l$, $l = 1, 2, \ldots, n_f$, is the Lagrangian force density $\mathbf{F}_b$ associated with the node l; n_f is the total number of grid points of the plate, the δ_{kl} is the Kronecker symbol whose definition is

$$\delta_{ml} = \begin{cases} 1, & \text{if } m = l, \\ 0, & \text{if } m \neq l. \end{cases}$$

The sums in Eqs. (16) and (18) can be simplified by making use of the Kronecker delta property, but it is actually better *not* to make this simplification, but to compute the forces directly from Eqs. (16) and (18), since this avoids complications at the four edges of the plate.

Note that Eq. (15) is a discretization of Eq. (6) along the longitudinal direction (i.e. α_1) and Eq. (16) is a discretization of Eq. (7) along the longitudinal direction (i.e. α_1). Eq. (16) and Eq. (18) are obtained by taking the derivative with respect to the variable $\mathbf{X}$ in Eqs. (15) and (17), respectively. Therefore Eqs. (16) and (18) give the force of a typical longitudinal fiber. The same calculation is repeated for all the longitudinal fibers (i.e. integration on α_2). The forces exerted by the lateral fibers are computed in a similar way except that now α_2 denotes the arc-length and α_1 denotes different lateral fibers.

Note that the total Lagrangian force density $\mathbf{F}(\boldsymbol{\alpha}, t) = \mathbf{F}_s(\boldsymbol{\alpha}, t) + \mathbf{F}_b(\boldsymbol{\alpha}, t)$. The two integral relations can be discretized as follows:

$$\mathbf{f}_{ib}^n(\mathbf{x}) = \sum_{\boldsymbol{\alpha}} \mathbf{F}^n(\boldsymbol{\alpha}) \delta_h(\mathbf{x} - \mathbf{X}^n(\boldsymbol{\alpha})) \Delta\boldsymbol{\alpha} \tag{19}$$

$$\mathbf{U}^{n+1}(\boldsymbol{\alpha}) = \sum_{\mathbf{x}} \mathbf{u}^{n+1}(\mathbf{x}) \delta_h(\mathbf{x} - \mathbf{X}^n(\boldsymbol{\alpha})) h^3. \tag{20}$$

Here the notation $\sum_{\alpha}$ means that the sum with respect to $\boldsymbol{\alpha}$ is taken over all of a discrete collection of points (first sum over all discrete points of a fiber in the form $\alpha_1 = m\Delta\alpha_1$, where m is an integer; then sum over all discrete fibers in the form $\alpha_2 = q\Delta\alpha_2$, where q is an integer), while the notation $\sum_{\mathbf{x}}$ means that the sum with respect to $\mathbf{x}$ is taken over all discrete points of the form $\mathbf{x} = (ih, jh, kh)$, where i, j and k are integers, h is the meshwidth. δ_h is a smoothed approximation of the Dirac δ function. In the IB method, δ_h is written in the following form:

$$\delta_h(\mathbf{x}) = h^{-3}\psi\left(\frac{x}{h}\right)\psi\left(\frac{y}{h}\right)\psi\left(\frac{z}{h}\right) \tag{21}$$

where h is the mesh spacing, $\mathbf{x} = (x, y, z)$, and ψ is chosen as

$$\psi(r) = \begin{cases} \frac{1}{4}\left(1 + \cos\left(\frac{\pi r}{2}\right)\right), & \text{if } |r| \le 2 \\ 0, & \text{otherwise.} \end{cases}$$

See [38] for details regarding the choice of $\psi(r)$. Note that the support of δ_h is a cube with width $4h$ at each discrete plate point instead of a sphere with diameter of $4h$. See [89] for an improvement to the discretization of the delta function to reduce force oscillations for moving-boundary simulations.

With $\mathbf{U}^{n+1}(\boldsymbol{\alpha})$ known from Eq. (9), the plate motion equations are discretized as follows:

$$\frac{\mathbf{X}^{n+1}(\boldsymbol{\alpha}) - \mathbf{X}^{n}(\boldsymbol{\alpha})}{\Delta t} = \mathbf{U}^{n+1}(\boldsymbol{\alpha}). \tag{22}$$

A summary of our algorithm is as follows: Given the values of all variables at the nth time step, we advance sequentially values of all variables at the $(n + 1)$th time step as follows:

(1) Compute the elastic force $\mathbf{F}^{n+1}$ from $\mathbf{X}^n$ using Eqs. (16) and (18).
(2) Spread the Lagrangian force density $\mathbf{F}^{n+1}$ onto the fluid lattice using Eq. (19),
(3) Compute the particle collision, i.e. calculate $\frac{1}{\tau}(g - g^0)$ in Eq. (10) via the equilibrium velocity distribution from Eq. (13),
(4) Update the velocity distribution function via streaming and external forcing using Eq. (10),
(5) Compute the new fluid velocity $\mathbf{u}^{n+1}$ using Eqs. (11) and (12),
(6) Interpolate the velocity of the immersed boundary $\mathbf{U}^{n+1}$ from the velocity $\mathbf{u}^{n+1}$ of the ambient fluid using Eq. (20),
(7) Update the position of the immersed boundary (i.e. compute $\mathbf{X}^{n+1}(\boldsymbol{\alpha})$ via Eq. (22)).

3. Application — simulation of a 3D viscous incompressible flow past an elastic plate tethered at its midline

3.1. The model problem

We consider generalizing a previously studied problem of a 2D viscous flow past a flexible sheet tethered at the midline to 3D (see Fig. 2). The rectangular channel flow is affixed with a right-handed orthogonal coordinate system with the x-axis in a downward flow direction and the y-axis points from front to rear. The quadrilateral inside the box represents a deformable flexible sheet where the dashed line represents the tethered midline of the sheet. The front and rear, left and right boundaries are rigid walls. The top and bottom are respectively open inflow and outflow boundaries. The fluid is homogeneous, viscous and incompressible. The rectangular elastic sheet is matted with longitudinal and lateral linear elastic fibers having identical mechanical properties and is modeled by a lattice of Hookean springs of finite stiffness with a given node on the lattice connected to it four nearest neighbors. The stretching/compression coefficients of the longitudinal and lateral fibers are identical. The numerical values used in the simulations make the sheet somewhat extensible (an inextensible sheet is infinitely stiff and would cause numerical instability of the simulations). The maximum relative change in the length is less than 10%. The degree of stretching depends on the specific values of the dimensionless parameters used for a simulation. The fibers along each direction have identical bending moduli, which can be varied in our simulations.

The flow channel is an unit square in its cross section and has a length of two units in the flow direction, and the elastic sheet has a width to length ratio of 1 to 2. The midline of the sheet is tethered by stiff springs to a fixed line segment from point (x_t, y_{1t}, z_t) to point (x_t, y_{2t}, z_t). The sheet width is $W = y_{2t} - y_{1t}$. The sheet length is $L = 2W$. The sheet is placed initially horizontally, i.e. on the y–z plane. In the IB method, the drag may be computed conveniently by summing the tension at all of the nodes (discrete points) on the midline. The stiffness of the virtual spring is chosen to be very large such that the maximum magnitude of the displacement of the midline nodes is small enough that the sheet midline is virtually fixed in the physical space.

The three important dimensionless parameters used in this work are listed in Table 1. Here V_0 is the inflow speed, L is the sheet length, W is the sheet width, ν is the fluid kinematic viscosity, ρ_0 is the fluid mass density, M is the sheet mass density, K_b is the sheet bending modulus. Other dimensionless parameters are as follows: the stretching/compression coefficient of the fibers $\hat{K}_s = 20$; that of the virtual springs used to tether the midline of the sheet $\hat{K}_{st} = 20$–40. Note that the sheet is made of fibers of identical mechanical properties, i.e. the values of $\hat{K}_b$ and $\hat{K}_s$ are constant everywhere on the sheet including the midline that is tethered.

Table 1
Dimensionless parameters used in the simulations.

Name	Definition	Range
Reynolds number (Re)	$\frac{V_0 W}{\nu}$	0.1–200
Dimensionless flexure modulus ($\hat{K}_b$)	$\frac{K_b}{\rho_0 V_0^2 W^4}$	0.0001–0.05
Dimensionless mass density ($\hat{M}$)	$\frac{M}{\rho_0 L W}$	1

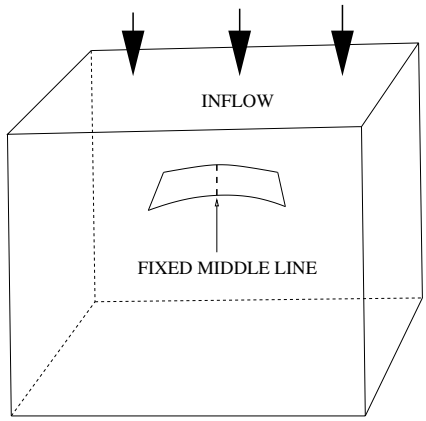

Fig. 2. A viscous flow past an elastic sheet tethered at the midline.

3.2. Validation of the numerical method

Convergence checking for a series of gradually refined grids ensures that the results obtained using the 3D IB/LBM method are reliable and, at the same time, it provides a measure of the accuracy. Note that when the grid size is refined, the physical time step is also halved, all the dimensionless parameters are fixed, and the numerical results are scaled back to the physical space (rather than in the lattice Boltzmann units). Simulations on grids of different sizes should run to the same physical time (the lattice Boltzmann steps are therefore usually different). Because of computational limitations, the convergence check was performed on a cubic channel, i.e. a channel with equal length, width, and height. The finest grid is taken as $80 \times 80 \times 80$; the coarsest grid is $10 \times 10 \times 10$. The mesh refinement ratio is 2. The results are as follows: When the IB is absent, the order of convergence for the first component of velocity (u) is nearly 2 in L_2 norm (the specific values are 1.6, 1.7), which agrees with the fact that the lattice Boltzmann method is second order in accuracy. When the IB is present, we get 1.2, 1.3 in L_2 norm for u which is reasonable because the formulation of the immersed 2D sheet in a 3D flow is singular, and that can cause order reduction near the immersed boundary. Similar results are obtained for the other two components of the velocity. The above numerical results indicate that our method is convergent and the order of accuracy is approximately 1 which is consistent with other versions of the IB method in general [6,36–38,46].

3.3. Flow and sheet visualization

The dynamics of the flow and of its interaction with the tethered sheet is visualized: A set of typical pictures is plotted in Figs. 3 and 4. The dimensionless parameters are as follows: $Re = 100, \hat{K}_b = 0.001$. The dimensionless time in lattice Boltzmann units is 119,600. Fig. 3(a) shows the streamlines of the flow. The gray surface represents the flexible sheet at the stationary state. Each thick curve represents a streamline whose tangent at any point represents the velocity direction at that point, and the color represents the magnitude of the velocity. Notice the complicated pattern right behind the flexible sheet (compared to other regions): the curves are tortuous and intertwined. This signifies the complexity of the flow field behind the sheet, i.e. the formation of vortices. It should be pointed out that this is a 3D plot and the seemingly intersecting curves do not actually intersect. Fig. 3(b) visualizes the velocity field via contours of the magnitude of the velocity. In this figure, different colors represent different values of the level set. The 3D space is cut open parallel to the x–y plane with the z-coordinate being 31.5 in lattice Boltzmann units. Fig. 4 visualize the vortical field through plotting the contours of the three components of the vorticity $\Omega = (\omega_1, \omega_2, \omega_3)$. Each level surface is identified by a distinct color. The attached color bar provides a measure of the level set's magnitude. The frame of the coordinates given in each figure gives the orientation of the flow. The 3D flow is visualized by cutting open three mutually orthogonal sectional planes: Fig. 4(a) gives the flow obtained by cutting open an x–y plane at $z = 40$; Fig. 4(b) gives the flow obtained by cutting open an x–z plane at $y = 31.5$; and, finally, Fig. 4(c) gives the flow obtained by cutting open an x–y plane at $z = 31.5$. From these figures we see that the vortex shedding is not intensive. Probably this is because the $Re = 100$ is not high enough. See Section 4 for a discussion.

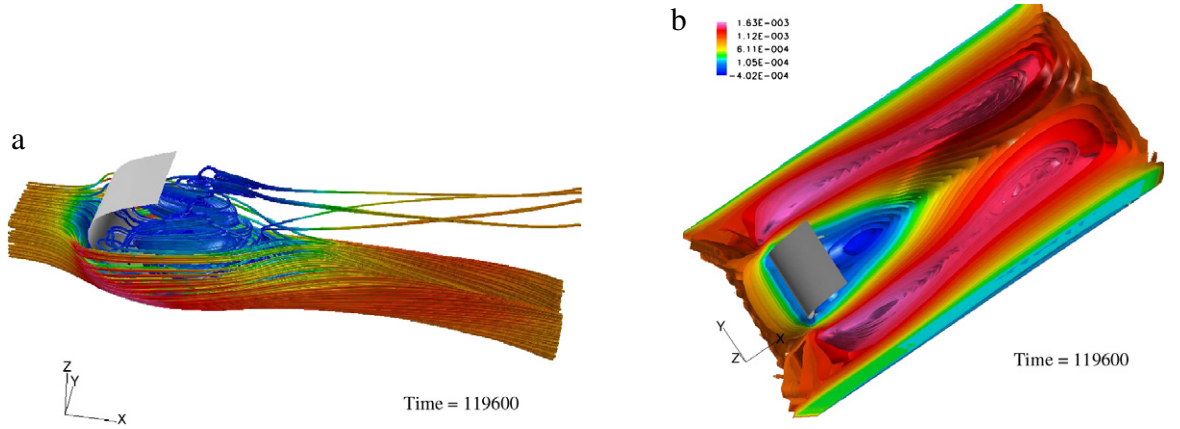

Fig. 3. Visualization of (a) the streamlines and (b) the velocity field via the contours of the velocity magnitude. Each color represents a different level set of the velocity magnitude.

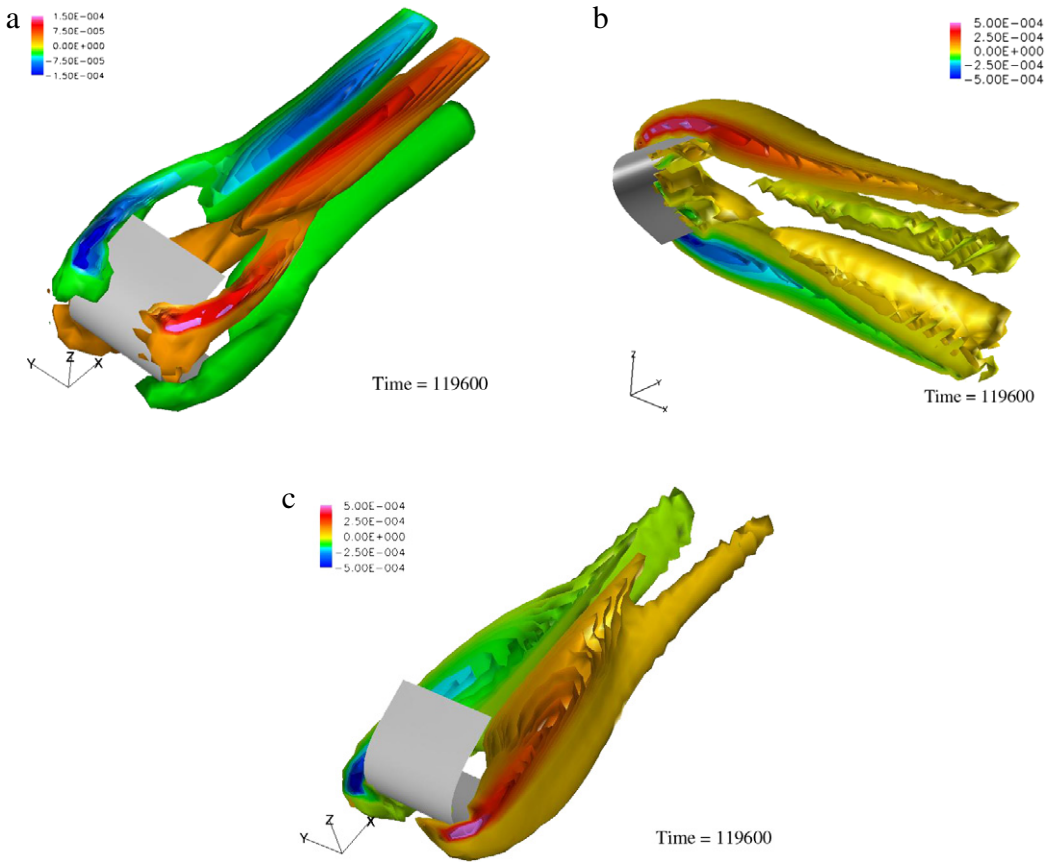

Fig. 4. 3D visualization of the vortical field via the contours of (a) ω_1, (b) ω_2 and (c) ω_3.

3.4. Drag scaling

The time-averaged drag coefficient is given by

$$\bar{C_d} = \frac{\bar{D}}{0.5\rho_0 V_0^2 LW},$$

where $\bar{D}$ is the time-averaged instantaneous drag of the sheet, L and W are the length and width of the sheet, respectively. The instantaneous drag D is recorded every t_r steps (the LB unit for time) from the initial time zero until the end of a simulation

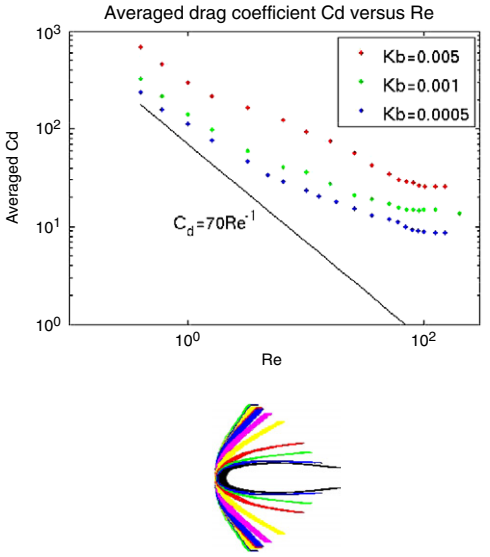

Fig. 5. Drag coefficient versus *Re* for three values of the bending modulus (top) and the shape and position of the sheet versus *Re* for a bending modulus of 0.005 (bottom).

(t_e). The averaging of D is done from an instant t_s when the flow reaches a quasi-steady state to t_e. In most of our simulations, $t_s = 50{,}000$, $t_e = 100{,}000$, and $t_r = 200$. Three series of simulations were performed with three different values of the elastic sheet's dimensionless bending modulus ($\hat{K_b} = 5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}$). In each series, the Reynolds number varies from 0.1 to 200, while the other dimensionless parameters are fixed.

3.4.1. Drag coefficient $\bar{C_d}$ versus Reynolds number Re

Plotted on a log–log scale in the top panel in Fig. 5 are three sets of time-averaged drag coefficient, $\bar{C_d}$ data as functions of *Re* for three different values of the dimensionless bending modulus $\hat{K_b}$. The red, green and blue solid dots represent respectively the time-averaged drag coefficient for $\hat{K_b} = 5 \times 10^{-3}$, $\hat{K_b} = 10^{-3}$, and $\hat{K_b} = 5 \times 10^{-4}$. From Fig. 5, we see that a more flexible sheet corresponding to a smaller value of $\hat{K_b}$ has a lower time-averaged drag coefficient than a less flexible one with a larger value of $\hat{K_b}$. This is expected because flexibility may induce drag reduction. For the three values of the bending modulus, the $\bar{C_d}$ data share a common feature: when the Reynolds number *Re* is sufficiently small, the drag coefficient $\bar{C_d}$ is approximately the reciprocal of the Reynolds number (the data corresponding to each $\hat{K_b}$ are approximately parallel to the line $C_d = \frac{k}{Re}$ where k is a constant) which is reasonable [85]; when *Re* is sufficiently large, $\bar{C_d}$ is nearly constant ($\bar{C_d}$ scales as the $zero^{th}$ power of *Re*); when *Re* is in between, the drag coefficient does not scale as a single power of *Re* (each set of the data does not fit to any single straight line).

Plotted on the bottom panel of Fig. 5 are the shape and position of the sheet after it reaches a quasi-steady state for different Reynolds numbers with dimensionless bending modulus of 5×10^{-3} (other values of $\hat{K_b}$ produce similar figures). Each color surface (looking like a curve when reviewed from a certain angle) identifies a sheet with a different Reynolds number. The innermost surface corresponds to the smallest *Re* and the outermost surface corresponds to the greatest *Re*. As *Re* increases the sheet position expands outwards. Note that some sheet positions and shapes are very close to each other and look as though they were overlapped in the figure. Intuitively, one may argue that as *Re* increases the greater inertial forces would cause the sheet to bend more and, thus, the sheet position should contract inwards. But this is not the case according to our simulations. A similar phenomenon was also found in two dimensions [83]. It appears that the fluid viscous forces are more influential as regards the bending of the sheet than the inertial forces. The smaller the *Re*, the greater the viscous force—thus causing the sheet to bend more.

3.4.2. Drag scaling and the sheet's flexibility η

We introduce a dimensionless drag $D = \bar{C_d}\eta^2$, and a second variable $\eta = \frac{1}{\sqrt{K_b}}$ to characterize the elastic sheet's flexibility. Note that a larger value of η implies a more flexible sheet while a greater value of $\hat{K_b}$ implies a stiffer sheet. It can be shown that D scales with η in the same way as the total dimensional drag scales with the inflow speed U. The dimensionless drag D is presented as functions of the sheet flexibility η for three different Reynolds numbers on a log–log scale in the top panel of Fig. 6. The red, black, and green bullets represent respectively results for $Re = 1$, 10, and $Re = 100$. The three straight lines (red, black and green) are the least square lines fitted to the data for a given value of *Re*. The slopes for the three lines are approximately 1.1, 0.9 and 1.0, corresponding to $Re = 1$, 10, 100, respectively. This suggests that the drag D as a

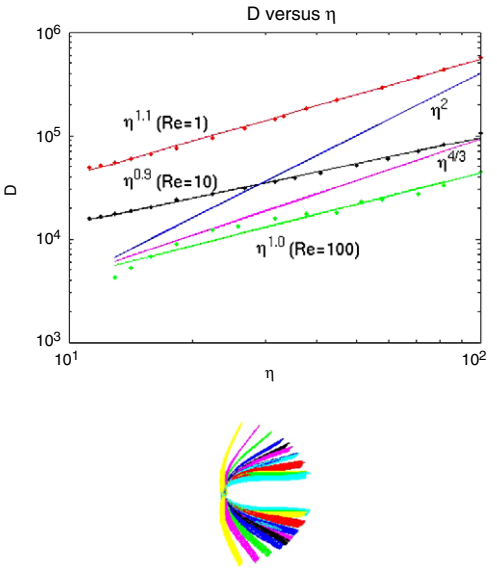

Fig. 6. Drag scaling for three values of the Reynolds numbers (left) and the shape and position of the sheet versus the bending modulus for $Re = 100$ (right).

function of flexibility η may be approximated as a power law for a given Re, and the exponent is approximately unity for Re in [1, 100]. It seems that the drag of a flexible sheet scales approximately as the inflow speed for Re between 1 and 100. Recall that in the case of a 2D viscous flow past a flexible fiber fixed at its midpoint the exponent of drag scaling (power law) decreases approximately from 2 towards 4/3 as Re increases from approximately 10 to 10^4. Our 3D simulation results show that flexibility induces more drag reduction at low Reynolds numbers. Note that a similar drag scaling was found by Qi [90] for the drag of multiple flexible chains falling in a viscous flow at low Reynolds numbers.

The bottom panel of Fig. 6 gives the shape and position of the sheet after it reaches a quasi-steady state for different values of the dimensionless bending modulus $\hat{K}_b$ with $Re = 10$ (similar figures for other values of Re). Each color surface (which may look like a curve when viewed from certain angles) corresponds to a sheet with a different value of $\hat{K}_b$. The innermost surface corresponds to the smallest $\hat{K}_b$ and the outermost surface corresponds to the largest $\hat{K}_b$. As $\hat{K}_b$ increases the sheet position moves outwards. This is because a more flexible sheet (with a lower $\hat{K}_b$ value) is more easily bent and streamlined by the local flow. Note that some sheet positions and shapes are rather close to each other and they look as though they were overlapped in the figure.

4. Summary and discussion

We have proposed a 3D immersed boundary (IB) method using the D3Q19 lattice Boltzmann model to represent the flow model. Our numerical study indicates that the hybrid method is first-order accurate, which is comparable to the situation for other versions of the IB method in general. The newly developed method is employed to simulate a viscous flow past a compliant sheet fixed at its midline in a 3D channel. Our numerical results indicate that the drag of the flexible sheet scales approximately as the inflow speed in the ranges of the dimensionless parameters used in the simulations. This is in sharp contrast with the drag of a rigid body in a viscous flow (drag scales as the square of the inflow speed).

Our numerical results for the flexible-sheet–fluid interaction are based on simulations with dimensionless parameters in certain ranges: $0.1 \leq Re \leq 200$, $0.0001 \leq \hat{K}_b \leq 0.05$. We would like to point out that when the Reynolds number Re or the dimensionless bending rigidity $\hat{K}_b$ are too large, the numerical method becomes unstable. When $\hat{K}_b$ is sufficiently large, small deformations of the immersed boundary may generate sufficiently large force to induce overshoot of the immersed boundary leading to instability. When the Reynolds number is sufficiently large, inertia forces dominate over viscous forces. This gives rise to rapid relaxation of the immersed solid boundary towards tangential equilibrium, which engenders an even stricter constraint on the time step size for an explicit IB method such as ours to maintain numerical stability. Moreover, as Re increases, flow gradients are more confined, i.e. thinner boundary layers formed, vortical flows are tighter spirals, etc. So, the flow variations grow larger. Spatial resolution deteriorates and a solution can grow unboundedly. Therefore finer spatial resolution and a smaller time step size are needed to correctly resolve the flow. The possible synergistic interaction of these two factors may render the stability analysis more difficult. Significant efforts along this direction are certainly needed in the future.

Note that except for the convergence check, the results presented here are based on simulations performed on a grid of $120 \times 60 \times 60$. The flexible sheet is 20×40 (width by height). Initially the spaces between the sheet edges and the walls are

10 along the z direction (top to bottom) and 20 along the y direction (front to rear). While the distance between the sheet edges and the walls in the y direction remains roughly the same, the distance along the z direction increases as the sheet bends with the flow. The final distance when the flow reaches a quasi-steady state varies from simulation to simulation depending on the specific parameters used. One may wonder what the wall effects on the simulation results are. The wall effects for the 2D version of this problem have been investigated quantitatively by Zhu [83]. We believe that the wall effects in 3D should be approximately the same in general. That is, compared to the unbounded case (the ratio of the sheet width to the channel width is zero), a finite size of the flow channel may delay the vortex formation and shedding and slightly increase the drag coefficient of the sheet.

In the case of a viscous flow past a rigid object, the vortex shedding is usually seen when Re is of the order of 100. But here in our case, vortex shedding is not apparent at this range of Re. Presumably this is because the sheet is deformable. The existence of a second dimensionless parameter, i.e. the flexure modulus $\hat{K}_b$, may quantitatively modify the role of Re. It has been shown in the 2D equivalent [83] of the 3D problem here in our work that the critical Re for vortex shedding increases as the $\hat{K}_b$ decreases, i.e., the flexibility of the sheet will delay the vortex shedding compared to that for a rigid sheet. Together with the similar effect of a finite flow channel, this may explain why the vortex shedding is not intensive for our simulations at Re round 100.

Notice that the drag coefficient in our case (a flexible sheet) is in general approximately one order of magnitude greater than that of a *rigid* plate placed normal to an incompressible viscous flow [91]. To explain this seeming contradiction, we would like to point out first that the flexible sheet in our simulations is in fact not inextensible: it stretches with the local flow and this can be seen in the bottom panel of Fig. 5. A stretched flexible sheet absorbs more elastic energy from the flowing fluid and therefore experiences more drag than an inextensible flexible body. Secondly, the drag reduction induced by body flexibility discussed in our paper means that the drag of a flexible body is reduced when it bends and streamlines with the local flow because of its flexibility. In our case it appears that the drag coefficient C_d decreases with the degree of body flexibility $\hat{K}_b$. This is because a more flexible sheet tends to bend more and represent itself as a smaller obstacle to the mainstream flow, and thus to reduce resistance from the flow. We hasten to add that this does not necessarily mean that the drag of a flexible body must be less than that of a corresponding rigid body of the same geometry. In some cases just the opposite is found. Here are two more examples. Existing laboratory experimental data [92,93] show that the drag coefficient of a rigid smooth plate aligned with a viscous flow is approximately 10 times less than that of a flexible stationary flag of the same geometry. A very recent work by [94] has found something similar: the drag coefficient of a flexible flapping biofilm is greater than that of a stiffer biofilm in a viscous flow. An intuitive explanation may be as follows: when a viscous fluid is moving past a flexible body, the local body surface, especially the edges of the body, may move more easily with the local flow compared to the rigid case. This kind of "extra" local movement of a flexible body which is absent in the case of a rigid body generates more disturbances to the local flow and thus causes more energy dissipation due to fluid viscosity. Consequently the flexible body may experience more resistance from the flow compared to a rigid equivalent. This effect becomes more important as the body becomes more flexible. This being said, however, analytical or quantitative results seems to be out of the question at this point. Certainly this issue is worthy of further study in the future.

Acknowledgements

The authors thank Dr. Raymond Chin for helpful discussions and careful proofreading of the manuscript. The authors also thank Dr. Z.L. Guo, Dr. B.C. Shi, and Dr. L.-S. Luo for discussions on the lattice Boltzmann method. The first author thanks the USA National Science Foundation (NSF) for support under the grant DMS-0713718.

References

[1] D. Weihs, Hydromechanics of fish schooling, Nature 241 (1973) 290–291.
[2] M.A.R. Koehl, The interaction of moving water and sessile organisms, Scientific American 247 (1982) 124–132.
[3] Y.C. Fung, Biomechanics: Mechanical Properties of Living Tissues, Springer, New York, 1993.
[4] X. Zhang, D. Schmidt, B. Perot, Accuracy and conservation properties of a three-dimensional unstructured staggered mesh scheme for fluid dynamics, J. Comput. Phys. 175 (2002) 764–791.
[5] X. Zhang, S.Z. Ni, G.W. He, A pressure-correction method and its applications on an unstructured Chimera grid, Comput. Fluids 37 (2008) 993–1010.
[6] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 25 (1977) 220.
[7] C.S. Peskin, The immersed boundary method, Acta Numer. 11 (2002) 479.
[8] T.J.R. Hughes, W. Liu, T.K. Zimmerman, Lagrangian–Eulerian finite element formulation for incompressible viscous flows, Comput. Methods Appl. Mech. Eng. 29 (1981).
[9] J. Donea, S. Giuliani, J.P. Halleux, An Arbitrary Lagrangian–Eulerian finite element method for transient dynamic fluid structure interactions, Comput. Methods Appl. Mech. Eng. 33 (1982) 689.
[10] D. Sulsky, Z. Chen, H.L. Schreyer, A particle method for history-dependent materials, Comput. Mech. Appl. Mech. Eng. 118 (1994) 179–197.
[11] D. Sulsky, S.J. Zhou, H.L. Schreyer, Application of a particle-in-cell method to solid mechanics, Comput. Phys. Commun. 87 (1994) 136–152.
[12] R. Glowinski, T. Pan, J. Periaux, A fictitious domain method for Dirichlet problem and applications, Comp. Methods Appl. Mech. Eng. 111 (1994).
[13] R. Glowinski, T. Pan, J. Periaux, A fictitious domain method for external incompressible viscous flow modeled by Navier–Stokes equations, Comp. Methods Appl. Mech. Eng. 112 (1994).
[14] R. Glowinski, T. Pan, T. Hesla, D. Joseph, J. Periaux, A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow, J. Comput. Phys. 169 (2001) 363.
[15] R.J. LeVeque, Z.L. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (1994) 1019–1044.

[16] R.J. LeVeque, Z.L. Li, Immersed interface methods for Stokes flows with elastic boundaries or surface tension, SIAM J. Sci. Comput. 18 (1997) 709–735.
[17] Z.L. Li, M.C. Lai, Immersed interface methods for Navier–Stokes equations with singular forces, J. Comput. Phys. 171 (2001) 822–842.
[18] Z.L. Li, The Immersed Interface Method–Numerical Solutions of PDEs Involving Interfaces and Irregular Domains, SIAM press, Philadelphia, 2006.
[19] X. Liu, R. Fedkiw, M. Kang, A boundary condition capturing method for Poisson's equation on irregular domain, J. Comput. Phys. 160 (2000) 151–178.
[20] M. Kang, R. Fedkiw, X. Liu, A boundary condition capturing method for multiphase incompressible flow, J. Sci. Comput. 15 (2000) 323–360.
[21] D. Nguyen, R. Fedkiw, M. Kang, A boundary condition capturing method for incompressible flame discontinuities, J. Comput. Phys. 176 (2002) 205–227.
[22] L. Zhang, A. Gersternberger, X. Wang, W.K. Liu, Immersed finite element method, Comput. Methods Appl. Mech. Eng. 193 (2004) 2051.
[23] W.K. Liu, D.K. Kim, S. Tang, Mathematical foundations of the immersed finite element method, Comput. Mech. (2005) doi:10.1007/s00466-005-0018-5.
[24] X. Wang, W.K. Liu, Extended immersed boundary method using FEM and RKPM, Comput. Methods Appl. Mech. Eng. 193 (12–14) (2004) 1305.
[25] X. Wang, From immersed boundary method to immersed continuum method, Int. J. Multiscale Comput. Eng. 4 (1) (2006) 127–145.
[26] X. Sheldon Wang, An iterative matrix-free method in implicit immersed boundary/continuum methods, Comput. Struct. 85 (2007) 739–748.
[27] T.Y. Hou, Z.L. Li, S. Osher, H.K. Zhao, A hybrid method for moving interface problems with application to the Hele-Shaw flow, J. Comput. Phys. 134 (1997) 236–252.
[28] G.H. Cottet, E. Maitre, A level set formulation of immersed boundary methods for fluid–structure interaction problems, C.R. Acad. Sci. Paris, Ser. I 338 (2004) 581–586.
[29] J. Xu, Z. Li, J. Lowengrub, H. Zhao, A level set method for interfacial flows with surfactant, J. Comput. Phys. 212 (2) (2006) 590–616.
[30] G.H. Cottet, E. Maitre, A level set method for fluid–structure interactions with immersed interfaces, Math. Models Methods Appl. Sci. 16 (2006) 415–438.
[31] H.S. Udaykumar, W. Shyy, M.M Rao, A mixed Eulerian–Lagrangian method for fluid flows with complex and moving boundaries, Int. J. Numer. Methods Fluids 22 (1996) 691.
[32] T. Ye, R. Mittal, H.S. Udaykumar, W. Shyy, An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries, J. Comput. Phys. 156 (1999) 209.
[33] R. Mittal, G. Iaccarino, Immersed boundary methods, Ann. Rev. Fluid Mech. 37 (2005) 239–261.
[34] I. Borazjani, L. Ge, F. Sotiropoulos, Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3D rigid bodies, J. Comput. Phys. 227 (16) (2008) 7587–7620.
[35] C.S. Peskin, Flow patterns around heart valves: a digital computer method for solving the equations of motion. Ph.D. Thesis. Physiol., Albert Einstein Coll. Med, Univ. Microfilms. 378: 72–30, 1972.
[36] C.S. Peskin, D.M. McQueen, Computational biofluid dynamics, Contemp. Math. 141 (1993) 161.
[37] C.S. Peskin, D.M. McQueen, A general method for the computer simulation of biological systems interacting with fluids, Sympos. Soc. Exp. Biol. 49 (1995) 265.
[38] C.S. Peskin, D.M. McQueen, Fluid dynamics of the heart and its valves, in: H.G. Othmer, F.R. Adler, M.A. Lewis, J.C. Dallon (Eds.), Case Studies in Mathematical Modeling: Ecology, Physiology, and Cell Biology, Prentice-Hall, Englewood Cliffs, NJ, 1996, p. 309.
[39] M.F. McCracken, C.S. Peskin, A vortex method for blood flow through heart valves, J. Comput. Phys. 35 (1980) 183–205.
[40] C.S. Peskin, B.F. Printz, Improved volume conservation in the computation of flows with immersed elastic boundaries, J. Comput. Phys. 105 (1993) 33.
[41] M.E. Rosar, C.S. Peskin, Fluid flow in collapsible elastic tubes: a three-dimensional numerical model, New York J. Math. 7 (2001) 281–302.
[42] A.M. Roma, C.S. Peskin, M.J. Berger, An adaptive version of the immersed boundary method, J. Comput. Phys. 153 (1999) 509–534.
[43] M.C. Lai, C.S. Peskin, An immersed boundary method with formal second order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2000) 705.
[44] D.M. McQueen, C.S. Peskin, L. Zhu, The immersed boundary method for incompressible fluid–structure interaction, in: Proceedings of the First M.I.T. Conference on Computational Fluid and Solid Mechanics, June 2001.
[45] B.E. Griffith, C.S. Peskin, On the order of accuracy of the immersed boundary method: Higher order convergence rates for sufficient smooth problems, J. Comput. Phys. 208 (1) (2005) 75–105.
[46] L. Zhu, C.S. Peskin, Simulation of a flexible flapping filament in a flowing soap film by the immersed boundary method, J. Comput. Phys. 179 (2) (2002) 452–468.
[47] Y. Kim, C.S. Peskin, Penalty immersed boundary method for an elastic boundary with mass, Phys. Fluids 19 (5) (2007), article number 053103.
[48] C. Tu, C.S. Peskin, Stability and instability in the computation of flows with moving immersed boundaries: a comparison of three methods, SIAM J. Sci. Statist. Comput. 13 (1992) 1361–1376.
[49] A.A. Mayo, C.S. Peskin, An implicit numerical method for fluid dynamics problems with immersed elastic boundaries, in: Fluid Dynamics in Biology: Proc. AMS-INS-SIAM Joint Summer Research Conf. Biofluiddynamics, Contemporary Mathematics, vol. 141, AMS, 1993, pp. 261–277.
[50] L.J. Fauci, A.L. Fogelson, Truncated Newton methods and the modeling of complex elastic structures, Comm. Pure Appl. Math. 46 (1993) 787.
[51] K. Taira, T. Colonius, The immersed boundary method: a projection approach, J. Comput. Phys. 225 (2) (2007) 2118–2137.
[52] Y. Mori, C.S. Peskin, Implicit second-order immersed boundary method with boundary mass, J. Comput. Phys. (2006) (submitted).
[53] P.J. Atzberger, P.R. Kramer, C.S. Peskin, A stochastic immersed boundary method for biological fluid dynamics at microscopic length scale, J. Comput. Phys. 224 (2) (2007) 1255–1292.
[54] P.J. Atzberger, P.R. Kramer, Error analysis of a stochastic immersed boundary method incorporating thermal fluctuations, Math. Comput. Simul. 79 (3) (2008) 379–408.
[55] Y.H. Qian, Lattice gas and lattice kinetic theory applied to the Navier–Stokes equations, Ph.D. Thesis. University Pierre et Marie Curie, Paris, 1990.
[56] S.Y. Chen, H.D. Chen, D. Martinez, W. Matthaeus, Lattice Boltzmann model for simulation of magnetohydrodynamics, Phys. Rev. Lett. 67 (1991) 3776.
[57] S. Hou, Lattice Boltzmann method for incompressible viscous flow, Ph.D. Thesis. Kansas State Univ., Manhattan, Kansas.
[58] X. He, L.-S. Luo, Theory of lattice Boltzmann method: from the Boltzmann equation to the lattice Boltzmann equation, Phys. Rev. E 56 (1997) 6811.
[59] X. He, L.-S. Luo, A priori derivation of the lattice Boltzmann equation, Phys. Rev. E 55 (6) (1997).
[60] L.-S. Luo, Unified Theory of the lattice Boltzmann models for nonideal gases, Phys. Rev. Lett. 81 (1998) 1618.
[61] X. He, S. Chen, R. Zhang, A lattice Boltzmann scheme for incompressible multiphase flow and its application in simulation of Rayleigh–Taylor instability, J. Comput. Phys. 152 (1999) 642–663.
[62] D.A. Wolf-Gladrow, Lattice-Gas Cellular Automata and Lattice Boltzmann Models — An Introduction, Springer, Berlin, 2000.
[63] L.-S. Luo, Theory of the lattice Boltzmann method: lattice Boltzmann models for nonideal gases, Phys. Rev. E 62 (4) (2000) 4982–4996.
[64] S. Succi, The Lattice Boltzmann Equation, Oxford Univ. Press, Oxford, 2001.
[65] S.Y. Chen, G.D. Doolen, Lattice Boltzmann method for fluid flows, Ann. Rev. Fluid Mech. 30 (1998) 329.
[66] Z.G. Feng, E.E. Michaelides, The immersed boundary-lattice Boltzmann method for solving fluid–particles interaction problems, J. Comput. Phys. 195 (2004) 602–628.
[67] Z.G. Feng, E.E. Michaelides, Proteus: a direct forcing method in the simulations of particulate flows, J. Comput. Phys. 202 (2005) 20–51.
[68] X.D. Niu, C. Shu, Y.T. Chew, Y. Peng, A momentum exchange-based immersed boundary-lattice Boltzmann method for simulating incompressible viscous flows, Phys. Lett. A 354 (2006) 173–182.
[69] A.J.C. Ladd, J. Fluid Mech. 271 (1994) 285.
[70] Y. Sui, Y.T. Chew, P. Roy, H.T. Low, A hybrid immersed-boundary and multi-block lattice Boltzmann method for simulating fluid and moving-boundaries interactions, Int. J. Numer. Methods Fluids 53 (11) (2007) 1727–1754.
[71] O. Filippova, S. Succi, F. Mazzocco, C. Arrighetti, G. Bella, D. Hanel, Multiscale lattice Boltzmann schemes with turbulence modeling, Comput. Sci. Eng. 170 (2) (2001) 812–829.

[72] H. Yu, S.S. Girimaji, L.S. Luo, DNS and LES of decaying isotropic turbulence with and without frame rotation using lattice Boltzmann method, J. Comput. Phys. 209 (2005) 599.
[73] Y. Peng, L.-S. Luo, A comparative study of immersed-boundary and interpolated bounce-back methods in LBE, Prog. Comput. Fluid. Dyn. 8 (1–4) (2008) 156–167.
[74] G. Le, J. Zhang, Boundary slip from the immersed boundary lattice Boltzmann models, Phys. Rev. E 79 (2009) 026701.
[75] Y. Sui, Y.T. Chew, P. Roy, Y.P. Cheng, H.T. Low, Dynamical motion of red blood cells in a simple shear flow, Phys. Fluids 20 (2008) 112106.
[76] Z.G. Feng, E.E. Michaelides, Robust treatment of no-slip boundary condition and velocity updating for the lattice-Boltzmann simulation of particulate flows, Comput. Fluids 38 (2009).
[77] Y. Cheng, H. Zhang, Immersed boundary method and lattice Boltzmann method coupled FSI simulation of mitral leaflet flow, Comput. Fluids (2010), in press (doi:10.1016/j.compfluid.2010.01.003).
[78] Z. Guo, C. Zheng, B. Shi, Discrete lattice effects on the forcing term in the lattice Boltzmann method, Phys. Rev. E 65 (4) (2002) 046308.
[79] V. Steinburg, Bend and survive, Nature 420 (2002) 473.
[80] S. Alben, M. Shelley, J. Zhang, Drag reduction through self-similar bending of a flexible body, Nature 420 (2002) 6915.
[81] S. Alben, M. Shelley, J. Zhang, How flexibility induces streamlining in a two-dimensional flow, Phys. Fluids 16 (5) (2002) 1694.
[82] L. Zhu, C.S. Peskin, Drag of a flexible fiber in a 2D moving viscous fluid, Comput. Fluids 36 (2007) 398–406.
[83] L. Zhu, Viscous flow past an elastic fibre tethered at its center point: vortex shedding, J. Fluid Mech. 587 (2007) 217–234.
[84] L. Zhu, Scaling laws for drag of a compliant body moving in an incompressible viscous fluid, J. Fluid Mech. 607 (2008) 387–400.
[85] G.K. Batchelor, An Introduction to Fluid Dynamics, Cambridge Univ. Press, Cambridge, 1967.
[86] P.L. Bhatnagar, E.P. Gross, M. Krook, A model for collision processes in gases, I: small amplitude process in charged and neutral one-component system, Phys. Rev. 94 (1954) 511.
[87] R. Mei, W. Shyy, D. Yu, L.S. Luo, Lattice Boltzmann method for 3-D flows with curved boundary, J. Comput. Phys. 161 (2000) 680–699.
[88] S.Y. Chen, D. Martinez, R.W. Mei, On boundary conditions in lattice Boltzmann methods, Phys. Fluids 8 (9) (1996) 2527–2536.
[89] X.L. Yang, X. Zhang, Z.L. Li, G.W. He, A smoothing discrete delta function approach to reduce force oscillations in moving boundary simulations, J. Comput. Phys. 228 (20) (2009) 7821–7836.
[90] D.W. Qi, Private communications, 2009.
[91] F.M Najjar, S.P. Vanka, Effects of intrinsic three-dimensionality on the drag characteristic of a normal flat plate, Phys. Fluids 7 (10) (1995) 2516–2518.
[92] A.C. Carruthers, A. Filippone, Aerodynamic drag of streamers and flags, J. Aircraft 42 (4) (2005) 976–982.
[93] A. Yoshizawa, Drag of a finite flat plate set parallel to a uniform flow, J. Phys. Soc. Japan 32 (1972) 1677.
[94] D. Taherzadeh, et al., Biofilms and drag, Biotechnol. Bioeng. 105 (3) (2010) 600–610.